Classroom 2000: Enhancing Classroom Interaction and Review

Gregory D. Abowd[1,2], Chris Atkeson[2], Ami Feinstein[2,3], Yusuf Goolamabbas[1,2], Cindy Hmelo[4], Scott Register[3], Nitin "Nick" Sawhney[1] and Mikiya Tani[1,5]

[1]GVU Center, [2]College of Computing,
[3]Office of Information Technology & [4]EduTech Institute,
Georgia Institute of Technology, Atlanta, GA, USA
{abowd, cga, yusufg, ceh, nitin, mik}@cc.gatech.edu,
{ami.feinstein, scott.register}@oit.gatech.edu
http://www.cc.gatech.edu/fce
[5]NEC Kansai C&C Research Laboratory, Osaka, JAPAN

ABSTRACT

The classroom is a site for rich CSCW activities. We are interested in prototyping future computing environments that will enhance the classroom experience and empower both student and teacher. In this paper, we describe the Classroom 2000 project at Georgia Tech, which is integrating pen-based technology, audio services and the World-Wide Web to support the capture of in-class activity for later review and to enhance group interactions. We describe the prototype classroom system and the results of an experiment using this technology in a lecture-based course.

Keywords

Classroom technology, pen-based computing, CSCW in education, using the WWW, audio indexing.

INTRODUCTION

Our vision for educational technology is to use it to empower both student and teacher by enhancing existing modes of classroom interaction and review. We can invent new modes of group and individual activity by breaking the physical and temporal boundaries of the traditional classroom and providing ubiquitous electronic access over time and space. We have begun a project to introduce and examine the effects of pen-based technology, recorded group interactions and the Web within a traditional lecture-based classroom. We call the project Classroom 2000 to suggest a futuristic approach that is not very far off in time. Our prototype classroom has been built over the last 7 months and tested in undergraduate and graduate computer science courses at Georgia Tech.

Imagine that while studying for an exam a student could query a repository of all information collected throughout the course. This would include intelligent content-based search through the teacher's prepared lecture notes, the student's own notes taken during class, and the audio and video records of the classes. In reviewing this information, the student could also make links between issues discussed in separate lectures. Then imagine that this retrieval and association could be done across all classes that an individual student had attended or all classes taught at an institution. Providing automated support for the capture, exploration and recreation of such a rich information source is the goal of this research.

With the availability of ubiquitous information technologies, such as the Web, most universities are able to provide students with access to vast repositories of educational materials. The dizzying number of Web pages devoted to classroom materials, from all disciplines, is a testimony to the value of these services, at least from the perspective of the instructor.[1] But ubiquitous computing in an educational setting is about more than just bringing the classroom, or more accurately a static view of the classroom experience, to the student.

For example, one of the interesting issues we would like to address is what will the impact be on the classroom when every student brings a personal digital assistant (PDA) or notebook computer to class? These computers will have wired and wireless connections to the campus network, allowing them to be used for communication as well as computation. Will these enviable resources be used to entertain students during dull lectures, to enhance current teaching approaches, or drive an evolution of new approaches to learning? Will we have to ask the students to put the machines away in order to preserve group interaction, or can we use these resources to enhance interaction? Will we still lecture to students, or will new forms of pedagogy evolve? Will we have formal class meetings at all, or can we use enhanced email and news groups to mediate some of the classroom interaction?

In order to teach more effectively and to build effective computer tutors, we would like to know what each student knows. By retaining these types of records of what is

GVU Technical Report, GIT-GVU-96-21.

1.See, for example, the World Lecture Hall Web page, URL: http://www.utexas.edu/world/lecture.

actually presented in classes for several years, lecturers will be able to prepare more effective presentations by efficiently looking at what actually happened in prerequisite classes. Lecturers could refer to previously presented concepts, and put more emphasis on ideas that had not been presented before. A student that is confused by the concept of "conservation of energy" can review the classes he took previously by asking the system to access previous times during which that concept was discussed.

Our aim is not to replace the traditional lecture-based style of pedagogy, at least not initially. We do feel that a lot of the information in a lecture is inefficiently recorded or lost. The long-range goal of Classroom 2000 is to make it easier for both teacher and student to benefit from the rich exchange of information that is possible in a classroom setting. We will move toward that goal by developing a suite of tools and a development methodology that allows us to build a software infrastructure for supporting the delivery, capture and review of educational materials.

Overview of paper

We will begin with a brief review of several areas related to the research in this paper. We will then describe the history and current state of the Classroom 2000 prototype, its overall system architecture to support different phases of the task, and implementation issues. We summarize the results of a quarter-long (10-week) experiment evaluating the effectiveness of the prototype in an actual classroom setting, with particular attention on its effect on group interaction within the class and how it supported after-class review of the classroom experience. We will conclude with a discussion of the future directions of the Classroom 2000 project and draw some general conclusions on our experience so far.

RELATED WORK

Shneiderman *et al* [11] discuss the effects of introducing technology into the classroom in terms of the paradigm shifts that result. All of the existing systems discussed in this article, and all of the attempts we are familiar with have some commonalities that we are trying to avoid. Technology in the hands of the student usually translates into a workstation at each desk. This approach is fine, even necessary, for classes which involve computer-based activities (such as programming). We want to investigate the usefulness of alternative interfaces which are less intrusive and allow natural handwritten notetaking, such as a pen-based laptops, PDAs or tablets.

Our work in Classroom 2000 has been greatly influenced by the work in ubiquitous computing and electronic notetaking done at Xerox PARC [15]. We want to capture information provided by the teacher during a lecture, so electronic whiteboard capabilities provided by the Xerox LiveWorks LiveBoard [3] naturally suggest themselves. The teacher in Classroom 2000 uses the Liveboard and our software to present and annotate prepared lecture presentations. The role of the LiveBoard in our prototypes has only relied on its ability to serve as a recording electronic whiteboard. We have not made use so far of the ability to connect the LiveBoard to other remote information sources (not even the student notebooks in the same class), as is done in the DOLPHIN project [12] and other similar efforts.

We also wanted to provide the students with an electronic notebook with the capability to take notes during the class that could be the basis for review after class. The Marquee note-taking prototype developed at Xerox PARC [14] came the closest to what we wanted to have in the hands of the students. Marquee provided a simple mechanism for producing notes with a pen-based interface that also created automatic indexing into a video stream. This intimate connection between the note-taking device and alternate information streams (audio and/or video) is a common theme, hinted at by work in ubiquitous audio at MIT's Media Lab [6] and work at Apple developing tools for personal annotated audio [2]. Our own interest in Classroom 2000 was to determine if automated support for creating audio-enriched notes provided enough value-added capability beyond paper-based notes to encourage student use. We are also interested in determining whether the audio indexing capability would alter the note-taking practices of the student, for better or worse.

Because the World-Wide Web has so transformed the way we distribute shared information, we decided to use it as the driving metaphor for most of the Classroom 2000 development. It is quickly becoming the norm for individual courses at many universities to have their own Web page that serves as a central clearing house for all course documentation. While this use of the Web has some obvious advantages for both instructor and student, it does not really take on an active role in assisting the learning and teaching experience. We wanted to view the whole classroom experience as a Web authoring task and provide ways to capture and relate information before, during and after an actual classroom session. This more active use of the Web infrastructure is in tune with some recent applications of WWW technology in education [7, 10, 4]. Our major contribution beyond this existing work is the concentration on in-class capture of information that is to be coordinated with other classroom information via the Web.

Many researchers investigate the effect of technology in education. There is an important distinction for research in this area, based on whether the research is focused on education or on technology. We have taken a technology focus in our work so far, and this focus is evident in the way we describe our work and evaluate its impact. We are working toward an infrastructure and we are trying not to change too much of the existing teaching practices. Those will evolve as the technology progresses. This is in contrast with a more educational focus, as demonstrated by Wan and Johnson [13] or by Guzdial *et al.* [5], in which the purpose of the research is to understand and inform theories on learning. In the wider arena of educational technology, there needs to be both forms of research.

THE CLASSROOM 2000 PROTOTYPE

In designing the prototype for Classroom 2000, we found it useful to divide the activities into three distinct phases—pre-production, in-class use and post-production. The following subsections describe each of these phases in detail, with accompanying figures from the prototype system provided to make the discussion more concrete.

Pre-production phase

In the lecture-based model of the classroom, we assume that the teacher does some preparation for each lecture. This pre-production phase can range from the preparation of a complete slide-based presentation that will be shown in class to a less formal preparation of notes that the teacher alone will use during the lecture. Any prepared materials that the teacher wishes to make available to students during the lecture must be transformed into a format that can reside on the student's electronic notepad and be presented on the Liveboard during class.

To begin with, we are only fully supporting presentation of static information within the lecture. That means we support lectures that include writing on a chalkboard/whiteboard or using overhead transparencies or slides, but we do not support the presentation of videos or other dynamic information, such as a computer simulation or live demonstration. Support for dynamic information is a future consideration.

We were very concerned with minimizing the impact on the preparation of lecture materials in Classroom 2000, mainly because most lecturers will already have some form of lecture material that they will want to reuse. The more we required a lecturer to recreate information, the less likely they were going to want to adopt the Classroom 2000 technology. We decided to adopt PostScript as the universal representation for lecture material. As long as the lecturer could produce the on-line lecture material as a PostScript file, we would be able to handle it. We adapted some public domain (Unix) filters for converting PostScript files into a sequence of image files (in either GIF or BMP format).

The final step in the pre-production phase was to load the LiveBoard and individual student electronic notebooks with the image files for the lecture. This was done using a simple DOS batch file.

In-class/live phase

In the classroom itself, the instructor used a LiveBoard, which is a PC with a pen-sensitive, 67-inch diagonal screen. The students used a variety of pen-based units, ranging from x86 digital tablets to a 486 palmtop PC. All units (LiveBoard and student notebooks) were running the Microsoft Windows for Pen Computing drivers on top of Windows 3.1. We designed and implemented a slide presentation and annotation application in Visual Basic, called ClassPad, to run on all units. We developed our own application because none of the existing applications we had available for the LiveBoard system allowed us to easily log pen events and convert the resulting annotated slides into a form (e.g., GIF) that was easily displayed on all Web browsers. Figure 1 shows a typical screen shot of the ClassPad application.

As the lecturer is presenting the lecture, she can mark the prepared slides with additional annotations, similar to how she might write on top of her prepared slides using an overhead projector and markers. The students have their own copy of the slides which they are free to browse and mark as they please during the lecture. Blank slides can also be created on the fly in order to insert additional notes that might not fit on the prepared slides. The interface was designed to be as intuitive as possible and we went through several iterations of the ClassPad application before going live with students in the experimental class.

Figure 1: The ClassPad application

The entire audio portion of the lecture is digitally recorded. As the slides are browsed and annotated, certain significant events (e.g., changing slides, annotating a slide with a continuous writing gesture) are logged and the inked annotations are saved to an intermediate file. Upon completion of the lecture, the ClassPad application is exited and all logs and annotations are saved.

It is important to stress that in designing the ClassPad application, we tried to introduce as little extra work as possible for both lecturer and student. We wrote a single application that was usable on both the LiveBoard and the handheld units.

Post-production

Once the lecture is complete, we enter the post-production phase. The purpose of the post-production phase is to support the student and lecturer in reviewing material across all classes. Figure 2 shows an excerpt of the class Web page that presents part of the syllabus for the course. The Lecture column for each class date shows one or two bullets. The leftmost bullet is a pointer to the annotated slides from the LiveBoard. The rightmost bullet is a pointer to all of the students' notes from that class. The lecturer's or student's notes for a given lecture can be viewed by selecting the appropriate bullet.

Viewing with a frame-compatible browser (e.g., Netscape 2.0) produces the three-panelled presentation shown in Figure 3. Browsers that are not frame-compatible will get a different form of presentation than shown here. The top panel shows a thumbnail overview of all of the slides for the given lecture. A single thumbnail image can be selected for magnified viewing in the lower righthand panel. The current slide is indicated in the top overview panel by reverse video highlighting, as shown in Figure 3. The lower lefthand panel reveals some information about the content of the slide (extracted from the underlying text of the prepared slides), a list of audio links and a link to a contents-based search page.

The entire review presentation is automatically generated

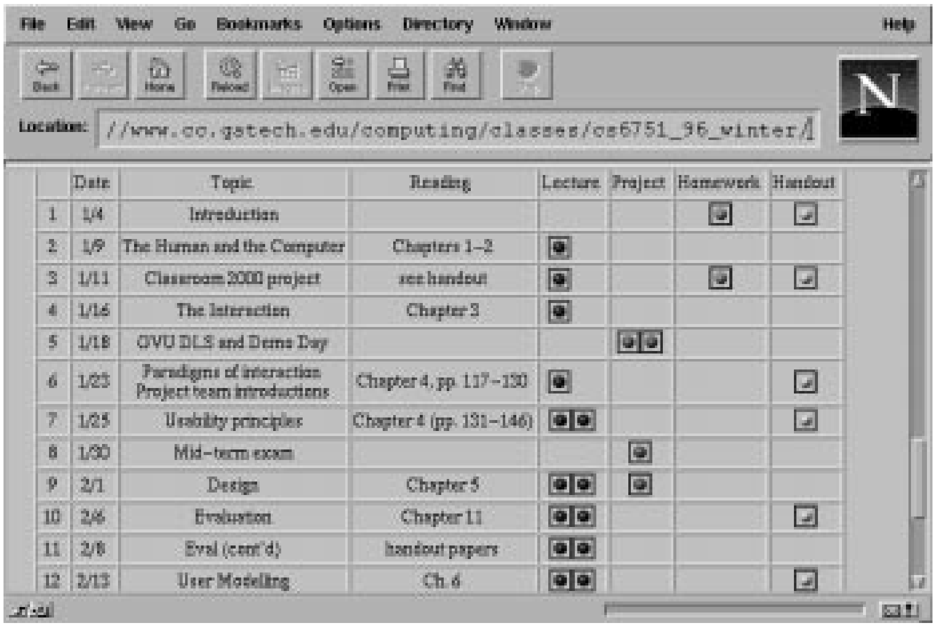

Figure 2: Class syllabus, indicating materials associated with each lecture

from the log files of a particular ClassPad session. We currently generate a timestamped audio link from a given slide for every visit to that slide during a class session. Selecting an audio link (indicated by timestamps underneath the Audio label in the lower left panel) invokes a custom-built audio player and sets it to the corresponding point in the audio stream for the given lecture. Once the audio client is playing the student can use the graphical interface to browse the audio track in either direction. The audio client communicates with a separate (also custom-built) audio server using streamed audio.

The audio annotated notes provides a value-added service beyond what students can achieve with paper-only notes. Without such value-added services, it is hard to justify the electronic notes. We provide one other value-added service in the current post-production phase, and that is the ability to do a limited content-based search on the slides across a whole course. Using the underlying text of the prepared slides, we create an inverted keyword index into the slides for a given lecture. These keywords are revealed at the top of the lower lefthand panel in Figure 3. By selecting the search link in this panel, the student will access a form through which they can submit simple textual queries that can span any number of lectures for a given course. For example, the student could search for all slides that contain the word “Evaluation” in them. The result of the search is another three-panelled review page, but the slides shown are not limited to a single lecture.

We will in the future be able to do some further post-processing of the collected lecture information (notes plus audio). For example, we could use a voice recognition system trained on the teacher’s voice to provide keyword-based search within and across lectures. We could use handwriting recognition to convert the notes into a more searchable form. We could create automatic links within and between lectures linking up parts of the course that discussed a common topic. These augmented capabilities would support both the student and the teacher in reviewing lecture and course material. Ultimately, the entire course would result in an on-line multimedia book authoring session.

System architecture and implementation issues

One of the critical insights early on in the project was the division of the problem into the three distinct phases of operation, as described above. The overall organization of Classroom 2000, what we refer to as the system architecture, provided a good separation of concerns. Developing tools that served activities in different phases of the project enabled concurrent development and is now allowing us to modify and enhance certain features of the system with minimal impact elsewhere.

Another important driving influence was to adhere as closely as possible to a Web-authoring metaphor. The Web excels at providing cross-platform access to a shared and distributed file system that holds information of a wide variety of types, and all of these features are important in the classroom setting. In addition, one of the important features of learning over a long period of time is how new information relates to old information. Whether this relationship results from explicit association done by the student during review or can be automatically inferred by the system, the basic operation that needs to be supported is the establishment of a link from some chunk of information to another.

Ultimately, we want to transition all phases of the project to activities completely supported by the Web and its associated browsers. In fact, we have already produced a prototype that does the live in-class annotation and time-stamping using client-side image maps. A pen-based interface to a browser is all that remains before we can completely transform the in-class phase to the Web.

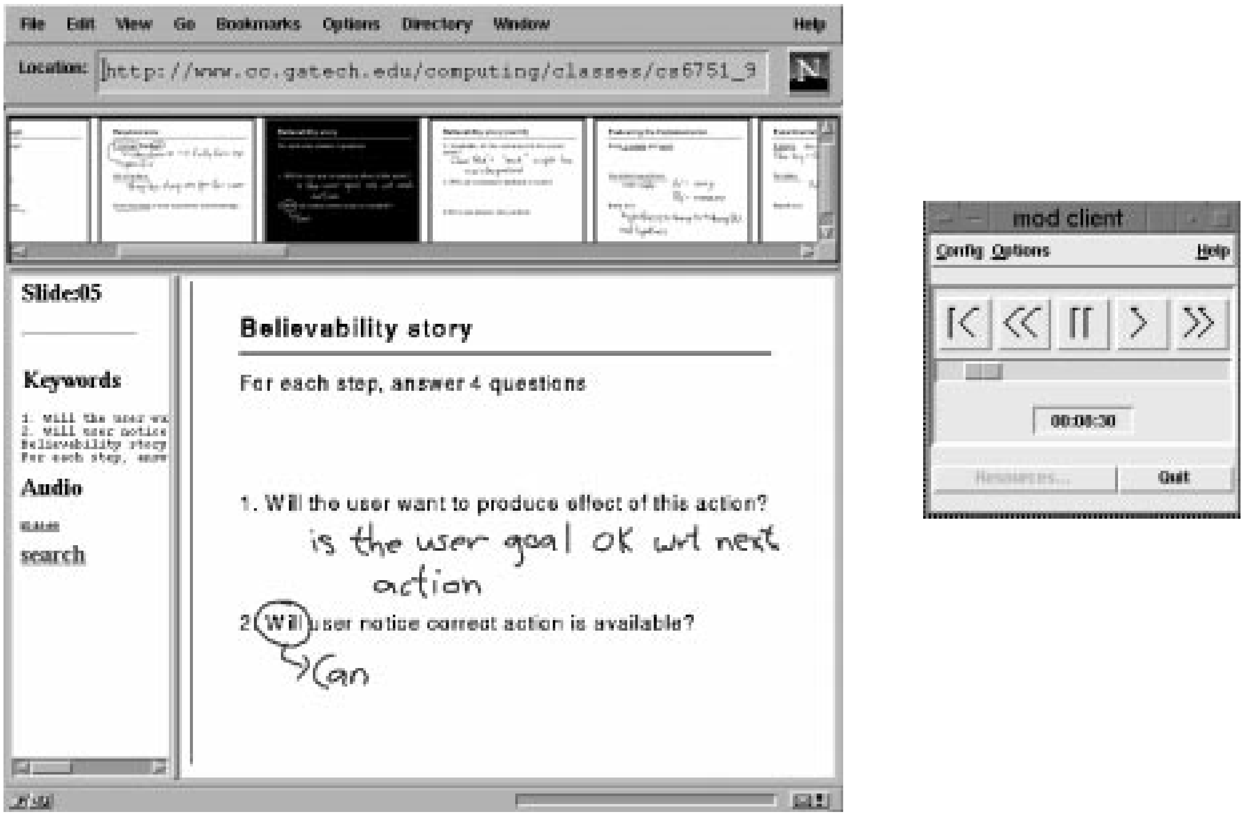

Figure 3: Review presentation (left) with accompanying audio client (right).

Another critical decision in our design was to model note-taking as an image annotation task. In many classroom settings, lecturers provide prepared slides or notes to the students before a class and the majority of students we polled considered this to be an advantage. In addition, many of the other tasks we perform in our daily lives are annotation tasks. Drafts of a paper are frequently annotated by reviewers and a teacher corrects a student's report by annotating it. So this annotation metaphor is a useful one for the classroom.

We also knew that anything drawn on top of an existing slide would lose its meaning if that slide was redisplayed in a different way (highly likely when using generic Web browsers on very different screen sizes). This registration problem can be avoided if the annotated image is a bitmap. The downside of this decision is that all file transfer (pre-production downloading, post-production uploading and image loading into a Web browser) are all slowed down. Also, the format of the review pages is less flexible in terms of how it can be displayed.

A final consideration in our design is that we ultimately want to be able to recreate the entire classroom experience. This means we will need to be able to handle richer media sources at a finer level of granularity. For example, the student should be able to ask during review, "What was the lecturer saying when I wrote this?" while pointing to some arbitrary annotation. Or the student might want to find the notes associated with a live demonstration that occurred at some point in the class. The solution we have produced for indexing and reviewing an audio stream for the class is immediately transferable to video. The obvious constraint at the moment is efficient storage and delivery of the richer media types. The granularity of the logging is too coarse at the moment and that is principally because of the limited programming interface of the Visual Basic environment used to develop ClassPad. As a demonstration prototype, Visual Basic has served us well, but it appears that we will have to move on to a more flexible environment.

These last concerns demonstrate the necessity to treat the in-class phase as a comprehensive capture activity. The bulk of the future Classroom 2000 activities will focus on post-production activities needed to reconstruct and filter the information.

EVALUATING EFFECTIVENESS

Initial prototypes

We developed all of the Classroom 2000 infrastructure during the Fall Quarter of 1995 at Georgia Tech. Throughout the quarter, different aspects of the prototype classroom were subjected to live evaluation in an undergraduate programming course with 40 students. We received constant feedback on various aspects of the prototype system (use of Liveboard, the ClassPad application, the Web-based syllabus and audio service).

Question	Regulars (4)	Occasionals (17)	Reluctants (4)
The technology in the class is something I would like to use in future classes	3.5	4.12	4.25
The technology in the class made me less worried about missing class.	3.5	3.88	3.75
The technology in the class made me less likely to miss class.	3.00	2.82	3.25
The technology in the class encouraged class participation	2.75	3.18	4.00
The technology in the class enhanced group discussions	3.00	3.06	3.5
The technology in the class impeded group discussions	2.33	2.35	2.00
The technology in the class allowed me to concentrate on understanding the material.	3.00	3.82	4.00

Table 1: Answers to select post-course questions

Almost all the initial reactions to new technology were positive, but that is to be expected amongst technically-oriented undergraduates. Students appreciated the increased class participation that the Liveboard encouraged and the increased availability and richness of lecture material after class (especially when they missed class). The organization of material on the Web greatly facilitated review. Several students chose to design and implement (in Smalltalk) a tool to help in the construction of the Web syllabus, making the task of maintaining the class page much easier for the teacher. Although the audio service was only completely functional for one lecture, students did find it useful to hear class discussions connected with static notes.

Full course experiment

During the 10-week Winter Quarter of 1996, the full Classroom 2000 prototype was used in a graduate computer science course in Human-Computer Interaction [1]. This was a project based class consisting of 25 students from a wide variety of disciplines across the Georgia Tech campus (10 from Computer Science, 4 masters students in a multidisciplinary program on Interactive Design of Technology, 4 from Electrical Engineering 2 from Chemistry, and one each from Textile Engineering, Earth and Atmospheric Sciences, Chemical Engineering, Aerospace Engineering and Industrial Engineering). Due to the budget for this project, we were unable to purchase enough units to provide each student with a pen-based electronic notebook. The number of units varied throughout the course from a minimum of 6 to a maximum of 12 working units. The technology in the class was phased in incrementally, beginning with the Liveboard, audio recording and, finally, student units. By the third week of class, students were taking electronic notes. Four students (the Regulars) were selected (from among 8 volunteers) to take notes using ClassPad for the duration of the quarter, which consisted of 10 lectures. Four students (the Reluctants) chose not to take notes electronically the entire class. The remaining students (the Occasionals) used the other units on a first-come, first-served basis, averaging 2.9 times each.

At the beginning of the course, the students filled out a questionnaire investigating their initial reactions to the use of the Classroom 2000 technology throughout the course. Students were asked to keep a journal of their handwritten notes, electronic notes (if they chose to print them out) and a lecture-by-lecture log of answers to a small questionnaire asking for their reactions to the use of the technology in the class that day and whether they had used the Web-based slides for review since the previous class. At the end of the course, the students filled out another questionnaire which investigated their overall reaction to the use of the technology in the class. Table 1 summarizes the students answers to a few of the questions in the final questionnaire that related directly to the effectiveness of Classroom 2000 in promoting group interaction.[2] In the above table, students were asked to give their reaction to a series of statements using the following scale: 1 (strongly disagree); 2 (disagree); 3 (neutral); 4 (agree); 5 (strongly agree).

The survey results show that all students would want to use this kind of technology in future classes. The availability of the LiveBoard notes with audio annotations made the students less worried about missing class, but didn’t actually encourage students to miss class. Three specific questions tried to determine to what extent the in-class technology was encouraging group interaction and discussion. The students who regularly used the ClassPad application were least positive about the impact the technology had on group interaction. When prompted for clarification, two factors emerged that seem to have caused this negative evaluation. First, the speed of response of the individual units was noticeably slower than that of the LiveBoard, and the students became frustrated with by delays. This decreased the likelihood that students would browse the slides during class because switching from one slide to the next was the most time-consuming operation in the system. Also, the students spent a lot of time copying verbatim what the lecturer wrote on the LiveBoard. The intent of the ClassPad application was to reduce the student’s burden for taking notes, but it appears not to have done so.

2.The final version of this paper will have a more complete analysis of the data, as the experiment just ended as this draft was being prepared.

For those students who only occasionally or never used the ClassPad application, there was indeed more of a feeling that the technology in the classroom freed them from some of the burden of taking notes and allowed them to concentrate on understanding the material in the lecture and participating in discussion. These students saw an advantage of the lecturer's notes being captured and available for later review. At this time, however, we have not been able to analyze these students' handwritten notes to see if they too are replicas of the lecturer's notes.

The students were asked whether they would give up paper-based note-taking techniques in favor of the pen-based electronic note-taking. Even with the usual objections that students don't have fast Internet connections in their homes or that reading a computer screen is hard on their eyes, nearly half of the students said that they would soon give up paper-based notes. Some of the reasons given were because the electronically accessible notes were more organized, searchable and more likely to grow in value over time as they are incorporated with notes from other classes. Several students noted that it was only after extended exposure to Classroom 2000 were they persuaded of its value.

RECOMMENDATIONS AND FUTURE WORK

There are some remedies to the problems which appear to be detracting from the group experience and overall effectiveness of Classroom 2000. In this section, we will address many of the problems we have discovered in our prototype classroom and discuss solutions that we are currently investigating.

The in-class phase operates in an entirely disconnected mode and it is clear that there are advantages to providing various levels of connection between the lecturer's notes and the student units and among the students themselves. Connection between the LiveBoard and the student units will enable students to selectively "copy" the lecturer's notes, potentially relieving the student of some of the burden of writing.

We relied on a single window interface to the ClassPad application. Whereas the resulting interface was intuitive and simple to use for making annotations, it is somewhat limiting. Access to multiple windows of information provides the ability to view the students notes concurrently with the lecturer's notes. Simple gestures can be supported to copy information from one view to another and to make hypertext links between different sets of notes. This can also be extended to provide students with the ability to view each other's notes in-class, though there are some more serious privacy issues that arise for this situation. Another advantage of the multiple view approach is that a student can pull up notes from a previous class. There were several occasions in the experimental class in which it would have been very useful to see portions of more than one lecture at a time.

An important issue that follows from the multiple windowing solution is one of screen real estate. The units we were using in class varied from a palmtop screen to a digital tablet. There were complaints about the size of the screen for the palmtops. We have considered providing students with multiple units during class to provide more viewing area. And since we are not operating under the assumption that the student units belong to the students, we can consider equipping a class with larger display devices, ultimately replacing the entire desktop with a pen-aware display.

The experimental class was project-based and all group projects centered around features of the Classroom 2000 prototype itself. There were seven project groups and each selected a project that fell into one of the phases of Classroom 2000 (pre-production, in-class and post-production). The majority of projects concentrated on improvements and extensions to the in-class note-taking application, providing capabilities for students to use keyword accelerators in their notes, (much like what is described in the Marquee system [14]), provide live, anonymous feedback to the instructor in the form of questions or opinions on the lecture topics, and to provide a finer level of granularity for timestamping notes to enhance the review mode.

One particular project proposed the use of two anonymous feedback devices from student to lecturer. The first used a stoplight/speedometer metaphor to allow the student to tell the lecturer whether they can proceed at the current pace, slow down or speed up. The second was a mood meter through which the student could indicate a level of interest in the current topic of discussion.

Many of the problems with this first experiment resulted from the lack of ubiquitous access to some features in the system. We were only able to provide a UNIX-based audio player and that greatly reduced the number of students who were able to access the audio recording. We are currently building a Java-based audio player that will provide cross-platform audio capabilities. We introduced the content-based search mechanism at the end of the class to help with the final exam and that was too late in the class to really assess its value. Also, the content-based search only operated over the prepared part of the lecture materials. It will be far more useful to provide content-based search facilities over the individual annotations (as is done by the Scribbler system [8]) or the audio stream (using speech recognition tools), and we are currently working toward this.

At the beginning of class, we decided to make all notes publicly available. This meant that anyone could read anyone else's notes; a fact that disturbed some of the students. Personal notes are not something that everyone wants to share. Students also complained that the reviewable notes were read-only and that they were not able to edit them after class. In fact, this was not the case; the HTML files were freely available for students to deal with as they pleased, but no one took advantage of this. It is clearer to us now that we need to be more proactive in getting the students to understand that they can preserve the privacy of their own electronic notes, maintain connection to the shared information from the class (lecturer's notes and the audio) and perform more sophisticated post-production tasks to modify their notes. We will also be concentrating more effort on providing tools to support post-production activities, such as linking concepts within and between lectures.

This paper has focussed on the use of pen-based interfaces to the exclusion of other input mechanisms. But there is no need for us to be so draconian. There are advantages to other input mechanisms, specifically the keyboard and paper, that the pen-based interface can at best only approximate. For example, some students noted that they can type faster than they can write and the typed in information is immediately available for content-based search mechanisms. We have talked about using handwriting recognition to translate penstrokes into text or employing pattern recognition techniques to feed a content-based search. But both of these solutions may be too heavyweight if keyboard input is convenient and preferred. We stayed away from keyboards because we felt the constant tapping of the keys was a distraction in the class. We also feel that purpose of Classroom 2000 is not to enable a student to take more notes, but rather to be more efficient. If we can automatically translate the live lecture into a textual transcript, then the student should not have to write down that much.

What about paper? It would be premature to suggest replacing paper entirely in the classroom, though we asked the students about this anyway. Paper is really good for some things. It is high resolution, reliable, portable, and inexpensive. With the improvements in vision research, and the subsequent applications of vision in projects such as the DigitalDesk at Xerox PARC [16], it is a reasonable assumption that we will soon be able to break down the barrier between the electronic world and the digital world.

The real thesis of Classroom 2000 is that providing rich capabilities for capturing classroom interaction for *all* students and lecturers will enhance the overall learning experience. Insufficient funds has currently limited us from testing that hypothesis but we are working on providing a more complete and cost-effective test-bed. For example, we are currently exploring cheaper alternatives to the LiveBoard with less functionality, such as a pen-based computer and a projection device. We are also concerned about the ramifications of this research on university policies which are considering requiring every student to own a personal computer. Just what kind of computer do we expect students to own? Should they have pen capabilities or not? One of the advantages of our approach is that we only rely on the pen-based computers as collection devices during the lecture. All post-production activities can take place on any machine that supports a Web browser.

Another intention of the project is to be able to completely recreate the classroom experience. Though we do manage to capture much of the information that is usually lost once the class ends, there is still much that we do not retain. We are aiming to be able to replay the entire lecture experience, including video and all student interactions with the ClassPad. Rather than challenge the student to recreate from memory the relevant information exchange from a class, we will challenge Classroom 2000 to filter the wealth of information that is captured to augment the individual student's memory of the group experience. We expect this to be a rich area of application for machine learning techniques, such as demonstrated by Schlimmer and Hermens [9]

The infrastructure provided by our Classroom 2000 prototype is ideal for other collaborative learning technologies. Specifically, it is an easy chore to link up the three-panelled review pages with scaffolding such as discussion groups provided by Guzdial's WebCamille system [5].

At Georgia Tech and elsewhere there is a strong emphasis on group work in classes. A lot of educational theories espouse the benefits of working in groups, yet we have not empowered the classroom to support small teams of students working together both during and outside of class. At this point, Classroom 2000 is a significant groupware application because it supports synchronous, co-located interaction, but the emphasis is on support for the individual and not for groups.

CONCLUSIONS

The goal of Classroom 2000 is to empower both student and teacher and enhance the learning experience that results from synchronous, co-located interactions of the traditional classroom lecture. We have defined a useful framework for conceptualizing the problem, dividing the problem into distinct phases representing pre-production, live in-class capturing activities and post-production/review. The clean separation of these phases is supported by the system architecture we have defined. The direct benefits of this architecture have been realized by our ability to easily modify components in one phase without affecting other phases. For example, we are currently building a version of ClassPad to run on a Newton MessagePad. This will be a completely new application that shares none of the development effort with the original ClassPad application developed in Visual Basic. Because we have defined a clear interface between the ClassPad application and the preproduction and post-production phases, we will be able to incorporate this new prototype into the whole system without much trouble.

We have implemented a solution to the preparation, presentation, capture and review of lecture-based material and have put that solution to the test for an entire graduate course in HCI. The results of our experiment are encouraging. Most students would want to continue with such technology in future classes and a good number are even convinced that they would forego traditional paper-based note-taking in favor of a more robust version of the prototype we provided. Our results also hold some warnings for enhancing group interaction. We have not provided a solution that improves the group learning experience. More accurately, we have provided a system that enhances the individual's record of a group interaction. Furthermore, it is our immediate goal to provide services that are more suited to assisting the individual, rather than enhancing group interaction.

We have only recently launched a research effort on Future Computing Environments research at Georgia Tech, and one of the main objectives of this research is to rapidly prototype interfaces that we imagine will be commonplace in 10-15 years. The Classroom 2000 project has gone from concept to longitudinal experimental study in under 7 months. We contend that this form of experimental research is essential given the pace of change of technology. We are not good at predicting the future capabilities and impact of technology. Instead, we should heed Alan Kay's advice and predict the future by inventing it. We should then evaluate it for its effectiveness.

Though we are stressing the rapid prototype of the near-future environment of Classroom 2000, we must always keep in mind the ultimate goals of the project. The standard of performance we seek for this system is to make each student feel as if they have had a personal tutor throughout their entire career. This tutor would know what the student knows, assist the student in linking previous knowledge to new knowledge, assist with group interactions with other students and teachers, and help the student understand and avoid previous mistakes. Once we meet this standard, we will no longer be speaking of the Electronic Classroom; rather, we will be referring to the Intelligent Classroom.

ACKNOWLEDGMENTS

Classroom 2000 has been a group development effort. The authors would like to acknowledge the support of the several members of the Future Computing Environments Group without whom this project would surely have failed. Those people are Savita Chandran, Rob Kooper, Sue Long and Dietmar Aust. Classroom 2000 is supported by a generous seed grant from the GVU Center at Georgia Tech. We would also like to thank Peter Freeman of the College of Computing, Janet Kolodner of the EduTech Institute and Ron Hutchins of the Office of Information Technology at Georgia Tech for their continued support of our work. Gregory Abowd would like to thank the wonderfully patient and insightful students in his Winter 1996 CS 6751 class for their help in providing an honest evaluation of the Classroom 2000 concept.

REFERENCES

1. G. Abowd. *Human–Computer Interaction*. Graduate Computer Science course offered at Georgia Tech, Winter 1996. Class home page is http://www.cc.gatech.edu/computing/classes/cs6751_96_winter.
2. L. Degen, R. Mander and G. Salomon. Working with Audio: Integrating Personal Tape Recorders and Desktop Computers. In *Human Factors in Computing Systems — CHI'92 Conference Proceedings*, pp. 413–418, 1992.
3. S. Elrod *et al.* Liveboard: A large interactive display supporting group meetings, presentations and remote collaboration. In *Human Factors in Computing Systems — CHI'92 proceedings*, pp. 599–607, May 1992.
4. J. Fowler, D. Baker, R. Dargahi, V. Kouramajian, H. Gilson, K. Brook Long, C. Petermann and G. A. Gorry. Experience with the Virtual Notebook System: Abstraction in Hypertext. In *Computer Supported Cooperative Work — CSCW'94 Conference Proceedings*, pp. 133–143, 1994.
5. M. Guzdial, *et al*., Computer support for learning through complex problem-solving. *Communications of the ACM*, 1996. In press.
6. D. Hindus and C. Schmandt. Ubiquitous Audio: Capturing Spontaneous Collaboration. In *Computer Supported Cooperative Work — CSCW'92 Conference Proceedings*, pp. 210–217, 1992.
7. M.-C. Lai, B.-H. Chen, S.-M. Yuan.Toward a New Educational Environment. In *International World Wide Web Conference — WWW'4 Proceedings*. http://www.w3.org/pub/Conferences/WWW4/Papers/238/. December, 1995.
8. A. Poon, K. Weber, and T. Cass. Scribbler: A Tool for Searching Digital Ink. In *Human Factors in Computing Systems — CHI'95 Conference Companion, Short Papers*. March 1995.
9. J. Schlimmer and L. Hermens. Software Agents: Completing Patterns and Constructing User Interfaces. *Journal of Artificial Intelligence Research 1*, pp. 61–89, 1993.
10. U. Schroeder, B. Tritsch, A. Krjerriem-Jasnoch. A Modular Training System for Education in the WWW Environment. In *International World Wide Web Conference — WWW'4 Proceedings*. http://www.w3.org/pub/Conferences/WWW4/Papers/306/. December, 1995.
11. B. Shneiderman *et al.* Windows of opportunity in electronic classrooms. *Communications of the ACM*, **38**(11):19–24, November 1995.
12. N. Streitz, J. Geissler, J. Haake and J. Hol. DOLPHIN: Integrated Meeting Support across Local and Remote Desktop Environments and LiveBoards. In *Computer Supported Cooperative Work — CSCW'94 Conference Proceedings*, pp. 345–358, 1994.
13. D. Wan and P. Johnson. Computer Supported Collaborative Learning Using CLARE: the Approach and Experimental Findings. In *Computer Supported Cooperative Work — CSCW'94 Conference Proceedings*, pp. 187–198, 1994.
14. K. Weber and A. Poon. Marquee: A tool for real-time video logging. In *Human Factors in Computing Systems — CHI'94 Conference Proceedings*, pp. 58–64, April 1994.
15. M. Weiser. The Computer for the 21st Century. *Scientific American*, September 1991.
16. P. Wellner. Interacting with paper on the DigitalDesk. *Communications of the ACM*, Vol. 36. No. 7, July 1993.
17. C. Wolf, J. Rhyne and L. Briggs. Communication and Information Retrieval with a Pen-based Meeting Support Tool. In *Computer Supported Cooperative Work — CSCW'92 Conference Proceedings*, pp. 322–329, 1992.
18. C. Wolf, J. Rhyne, L. Zorman and H. Ossher. WE-MET (Window Environment-Meeting Enhancement Tools). In *Human Factors in Computing Systems — CHI'91 Conference Proceedings*. pp. 441–442, 1991.